PROJECT REPORT ON WEB PHISHING DETECTOR

Domain : Applied Data Science

Team ID :
IBM-Project-2238-1658467898

College Name : *Panimalar Engineering College*

JANET RAJAJOTHI S
(211419106110)
Department of Electronics and Communication Engineering

KRITHIKA V
(211419106141)
Department of Electronics and Communication Engineering

KEERTHI G
(211419106132)
Department of Electronics and Communication Engineering

CHANDHINI R
(211419106055)
Department of Electronics and Communication Engineering

TABLE OF CONTENTS

1. INTRODUCTION

1.1. Project Overview

Phishing is a form of the attacker tries to learn sensitive information such as login credentials or account information by sending as a reputable entity or person in email or other communication channels.Typically a victim receives a message that appears to have been sent by a known contact or organization. The message contains malicious software targeting the user's computer or has links to direct victims to malicious websites in order to trick them into divulging personal and financial information, such as passwords, account IDs or credit card details.Phishing is popular among attackers, since it is easier to trick someone into clicking a malicious link which seems legitimate than trying to break through a computer's defense

1.2 PURPOSE

We have developed our project using a website as a platform for all the users. This is an interactive and responsive website that will be used to detect whether a website is legitimate or phishing. This website is made using different web designing languages which include HTML, CSS, JavaScript and Django.

The basic structure of the website is made with the help of HTML. CSS is used to add effects to the website and make it more attractive and user-friendly. It must be noted that the website is created for all users, hence it must be easy to operate with and no user should face any difficulty while making its use. Every naive person must be able to use this website and avail maximum benefits from it.

The website shows information regarding the services provided by us. It also contains information regarding ill- practices occurring in todays technological world. The website is created with an opinion such that people are not only able to distinguish between legitimate and fraudulent website, but also become aware of the mal-practices occurring in current world. They can stay away from the people trying to exploit ones personal information, like email address, password, debit card numbers, credit card details, CVV, bank account numbers, and the list goes on.

The dataset consists of different features that are to be taken into consideration while determining a website URL as legitimate or phishing.

2. LITERATURE SURVEY

2.1. Existing problem

Phishing is a technique used by hackers to fool internet users reveal their sensitive information like passwords, credit card numbers, contact information, and address, etc. Web phishing is carried out mostly by sending fake web links to the users through different communication means like Email, Facebook Messenger and WhatsApp, etc. Web phishing detection is significant for making internet browsing safe and secure for users. Different approaches were applied for the detection of fake websites. However, the most efficient method for detecting phishing websites is the one that is based on artificial intelligence and learning mechanism. In this research, an efficient and accurate method is proposed for the detection of phishing websites which is based on computational intelligence. Through the development of different computational models and rigorous testing, it was revealed that Extreme Gradient Boost (XGBoost) based model achieved the maximum scores in all the validation tests. This shows that the model is robust and accurate in terms of web-phishing detection.

2.2. References

a

1. J. Shad and S. Sharma, A Novel Machine Learning Approach to Detect Phishing Websites Jaypee Institute of Information Technology, pp. 425430, 2018.

2. Y. Sönmez, T. Tuncer, H. Gaskell, and E. Avci, Phishing web sites features classification based on extreme learning machine, 6th Int. Symp. Digit. Forensic Secure. ISDFS 2018 – Proceeding, vol. 2018 January, pp. 15, 2018.

3. T. Peng, I. Harris, and Y. Sawai, Detecting Phishing Attacks Using Natural Language Processing and Machine Learning, Proc. – 12th IEEE Int. Conf. Semant. Compute. ICSC 2018, vol. 2018Janua, pp. 300301, 2018.

4. M. Karabatan and T. Mustafa, Performance comparison of classifiers on reduced phishing website dataset, 6th Int. Symp. Digit. Forensic Secure. ISDFS 2018 – Proceeding, vol. 2018Janua, pp. 15, 2018.

5. S. Parekh, D. Parikh, S. Kotak, and P. S. Sankhe, A New Method for Detection of Phishing Websites: URL Detection, in 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, vol. 0, no. Icicct, pp. 949952.

6. K. Shima et al., Classification of URL bitstreams using bag of bytes, in 2018 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), 2018, vol. 91, pp. 15.

7. W. Fadhil, M. Abusharkh, and I. Abdel-Qader, On Feature Selection for the Prediction of Phishing Websites, 2017 IEEE 15th Intl Conf Dependable, Auton. Secure. Compute. 15th Intl Conf Pervasive Intel. Compute. 3rd Intl Conf Big Data Intell. Comput. Cyber Sci. Technol. Conger., pp. 871876, 2017.

8. X. Zhang, Y. Zeng, X. Jinn, Z. Yan, and G. Geng, Boosting the Phishing Detection Performance by Semantic Analysis, 2017.

9. L. MacHado and J. Gadge, Phishing Sites Detection Based on C4.5 Decision Tree Algorithm, in 2017 International Conference on Computing, Communication, Control and Automation, ICCUBEA 2017, 2018, pp. 15.

10. A. Desai, J. Jatakia, R. Naik, and N. Raul, Malicious web content detection using machine leaning, RTEICT 2017 – 2nd IEEE Int. Conf. Recent Trends Electron. Inf. Commun. Technol. Proc., vol. 2018Janua, pp. 14321436, 2018.

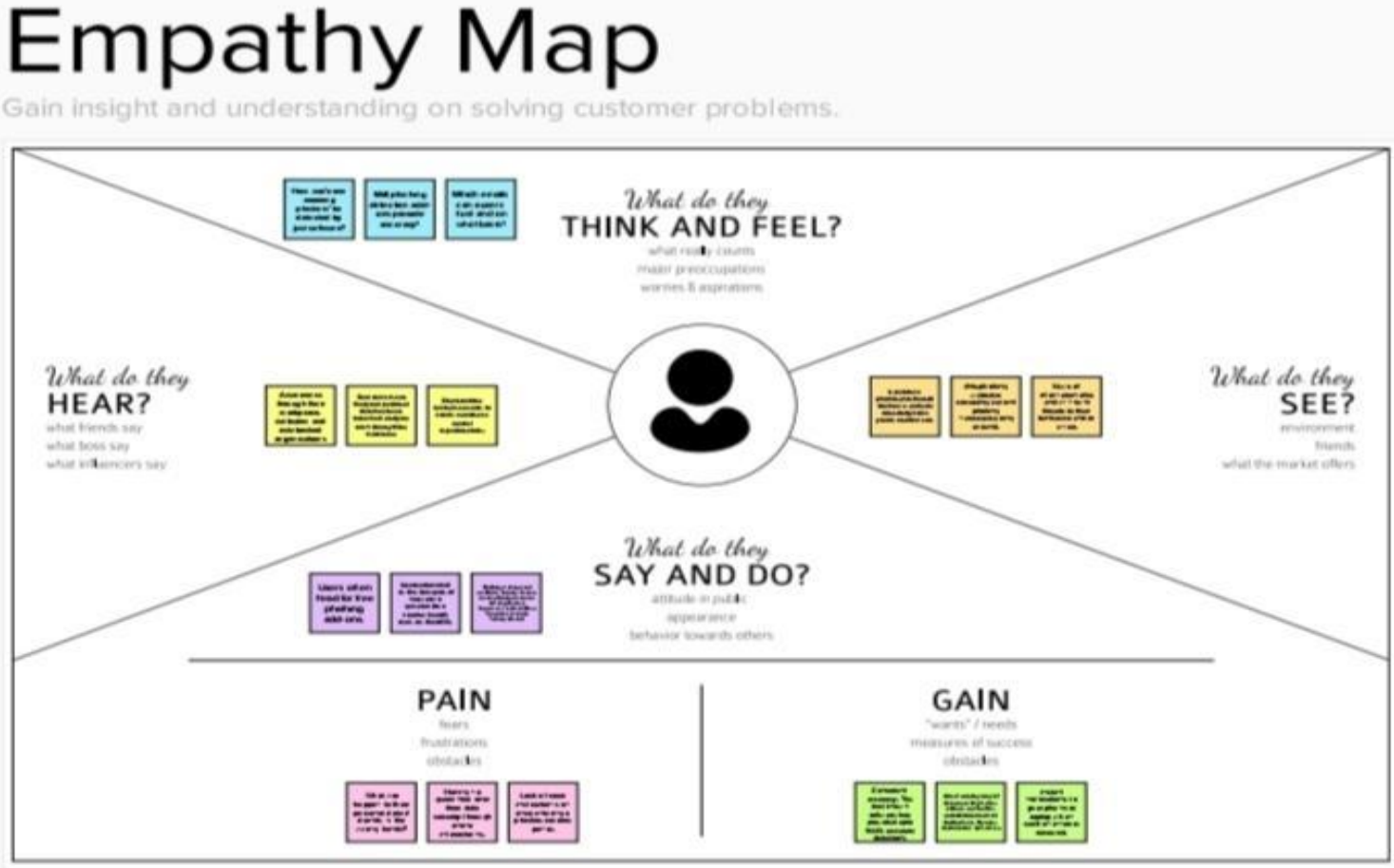
2.3. Problem Statement Definition

Problem Statement (PS)	I am	I'm trying to	But
PS-1	Internet user	**Browse the internet**	**I identify a scam.**
PS-2	Enterpriser user	**Open emails in the cloud server**	**I detect malicious protocols**

Problem Statement (PS)

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP



3.2 Ideation & Brainstorming

PRIORITIZE CHART:

3.3 Proposed Solution

1.Problem Statement (Problem to be solved)

Web phishing tends to steal a lots of information from the user during online transaction like username, password, important documents that has been attached to that websites. There are Multiple Types of Attacks happens here every day, but there is no auto detection Process through Machine Learning is achieved

2.Idea / Solution description

Through ML and data mining techniques like classification algorithm user can able to attain a warning signal to notify these phishing websites which helps the user to safeguard their identities and their login credentials etc. python is thelanguage that helps to enable these techniques for the online users.

3.Novelty / Uniqueness

This project not only able to identify the malicious websites it also has the ability to automatically block these kind of websites completely in the future when it has been identified and also blocks some various mails /ads from these malicious websites

4.Social Impact / Customer Satisfaction

This web phishing detection project attains the customer satisfaction by discarding variouskinds of malicious websites to protect their privacy. This project is not only capable of using by an single individual ,a large social community and a organisation can use this web phishing detection to protect their privacy. This project helps to block various malicious websites simultaneously.

5.Business Model (Revenue Model)

This developed model can be used as an enterprise applications by organisations which handles sensitive information and also can be sold to government agencies to prevent the loss of potential important data.

This project is not only capable of using by an single individual ,a large social community and a organisation can use this web phishing detection to protect their privacy. This project helps to block various malicious websites simultaneously.

6. Scalability of the Solution

This project is not only capable of using by an single individual ,a large social community and a organisation can use this web phishing detection to protect their privacy. This project helps to block various malicious websites simultaneously.

3.4 Problem Solution fit

Project Title: Web Phishing Detection

Project Design Phase-I - Solution Fit Template

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

An internet user who is willing to shop products online.

An enterprise user surfing through the internet for some information.

6. CUSTOMER CONSTRAINTS CC

Customers have very little awareness on phishing websites.

They don't know what to do after losing data.

5. AVAILABLE SOLUTIONS AS

Which solutions are available

The already available solutions are blocking such phishing sites and by triggering a message to the customer about dangerous nature of the website.

But the blocking of phishing sites are not more affective as the attackers use a different/new site to steal potential data thus a AI/ML model can be used to prevent customers from these kinds of sites from stealing data

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

The phishing websites must be detected in a earlier stage .

The user can be blocked from entering such sites for the prevention of such issues.

9.PROBLEM ROOT CAUSE RC

The hackers use new ways to cheat the naive users.

Very limited research is performed on this part of the internet.

7. BEHAVIOUR BE

The option to check the legitimacy of the Websites is provided.

Users get an idea what to do and more importantly what not to do.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

A trigger message can be popped warning the user about the site.

Phishing sites can be blocked by the ISP and can show a "site is blocked" or "phishing site detected" message.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?

The customers feel lost and insecure to use the internet after facing such issues.

Unwanted panicking of the customers is felt after encounter loss of potential data to such sites.

10. YOUR SOLUTION SL

An option for the users to check the legitimacy of the websites is provided.

This increases the awareness among users and prevents misuse of data, data theft etc.,

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

Customers tend to lose their data to phishing sites.

8.2 OFFLINE

Customers try to learn about the ways they get cheated from various resources viz., books, other people etc.,

Identify strong TR & EM

4.REQUIREMENT ANALYSIS

4.1Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Website identification	Model Detects the malicious website using blacklist And give alert message .This indicates that the website is encrypted and secured with an SSL (Secure Sockets Layer) certificate.
FR-4	Prediction	**prediction website** is Model predicts the URL using Machine Learning algorithms. such as Peredi On - Sports Prediction HTML Template, Tipster Platform HTML
FR-5	Classifier	Where people post there advertisement ,However, you can decide if you want the people to post for free or charge for their advertisements. Model predicts all the output to classifier and produces the final result.
FR-6	result	Model predict the website and give pop- up to the user before they enter any confidential details.

4.2 Non-functional Requirements:

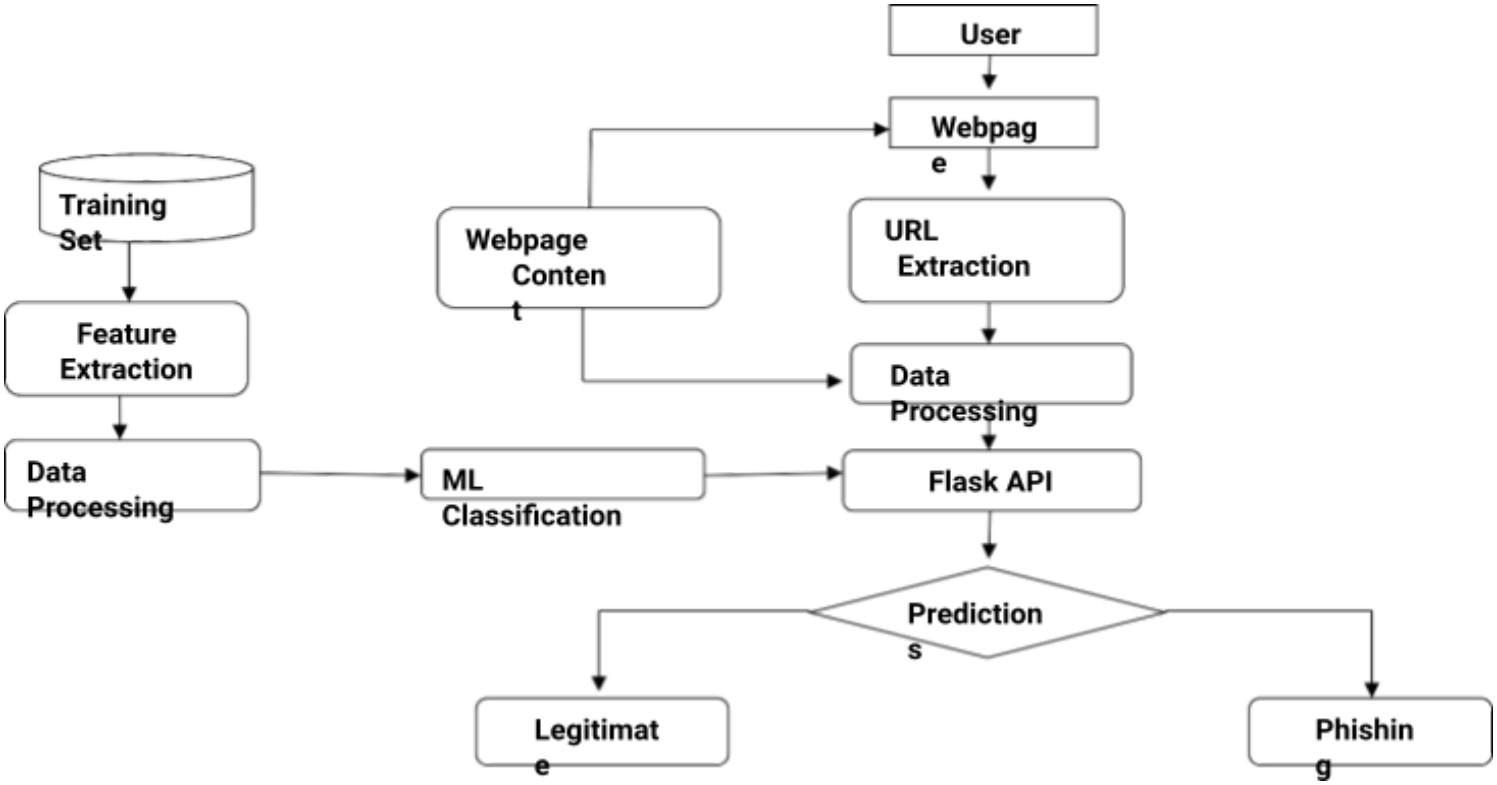
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	We intend to evaluate important **usability** issues related to **fake website** detection systems, 11, 13 . User can access to several website easily using web phishing detection without loosing any data
NFR-2	**Security**	User can check whether the websites are secure or not by getting pop-up message. Phishing is most often seen in the form of malicious emails pretending to be from credible sources like people, departments, or organizations related to the university.
NFR-3	**Reliability**	The users should get availability to access the resources must be valid and reliable
NFR-4	**Performance**	**performance** (i.e. detection and false positive rates) of the automated approaches. performance

		should be faster and user friendly for the effective performance
NFR-5	**Availability**	The users should get availability to access the resources must be valid and reliable.
NFR-6	**Scalability**	The performance of the website should be efficient to handle the increasing user and loads without any disturbance

5.PROJECT DESIGN

5.1Data Flow Diagrams



5.2 Solution & Technical Architecture Solution Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions.

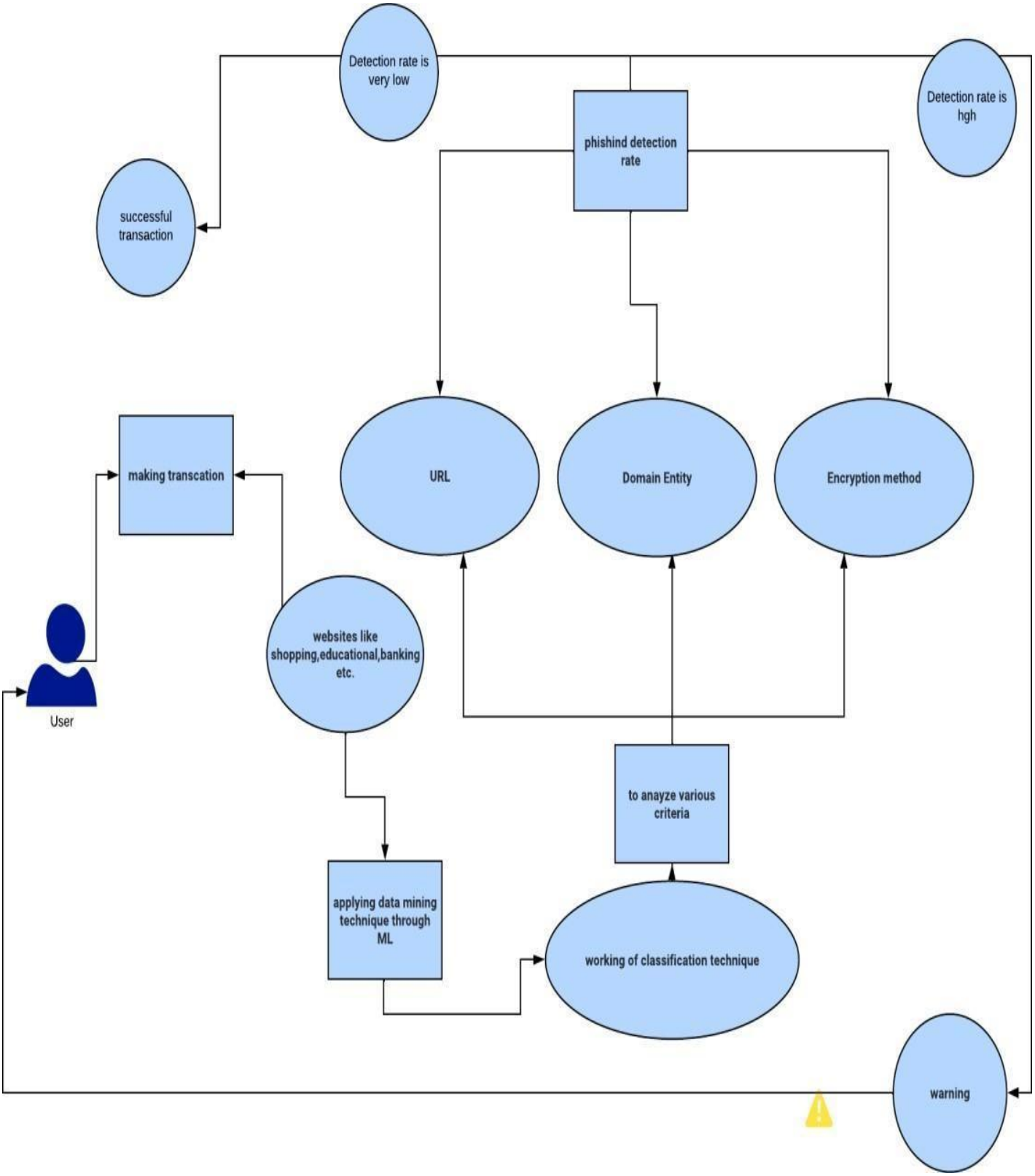
Its goals are to:

· Find the best tech solution to solve existing business problems.

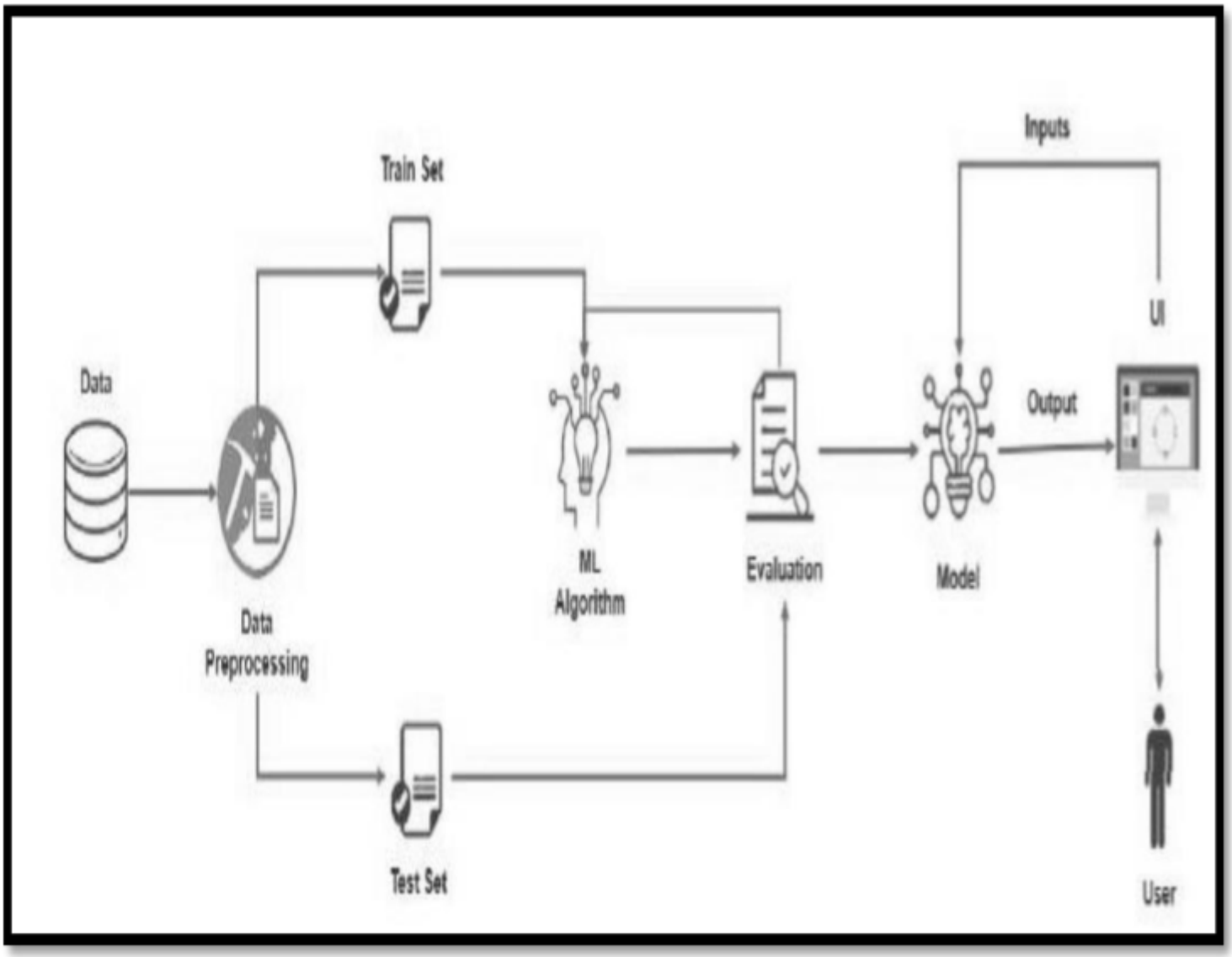
· Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.

· Define features, development phases, and solution requirements.

· Provide specifications according to which the solution is defined, managed, and delivered


Detection rate is very low
Detection rate is hgh
phishind detection rate
successful transaction
URL
Domain Entity
Encryption method
making transcation
websites like shopping,educational,banking etc.
User
to anayze various criteria
applying data mining technique through ML
working of classification technique
warning

Technical Architecture



5.3 User Stories

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	story points
Sprint-1	User input	USN-1	User inputs an URL in the required field to check its validation.	Medium	1
Sprint-1	Website Comparison	USN-2	Model compares the websites using Blacklist and Whitelist approach.	High	1
Sprint-2	Feature Extraction	USN-3	After comparison, if none found on comparison then it extract feature using heuristic and visual similarity.	High	1
1Sprint-2	Prediction	USN-4	Model predicts the URL using Machine learning algorithms such as logistic Regression, KNN.	Medium	1
Sprint-3	Classifier	USN-5	Model sends all the output to the classifier and produces the final result.	Medium	1
Sprint-4	Announcement	USN-6	Model then displays whether the website is legal site or a phishing site.	High	1
Sprint-4	Events	USN-7	This model needs the	High	1

			capability of retrieving and displaying accurate result for a website.	

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	team members story	
Sprint-1	User input	USN-1	User inputs an URL in the required field to check its validation.	Medium	Janet	1
Sprint-1	Website Comparison	USN-2	Model compares the websites using Blacklist and Whitelist approach.	High	krithika	
Sprint-2	Feature Extraction	USN-3	After comparison, if none found on comparison then it extract feature using heuristic and visual similarity.	High	chandhini	1
1Sprint-2	Prediction	USN-4	Model predicts the URL using Machine learning algorithms such as logistic Regression, KNN.	Medium	keerthi	1
Sprint-3	Classifier	USN-5	Model sends all the output to the classifier and produces the final result.	Medium	janet	1
Sprint-4	Announcement	USN-6	Model then displays whether the website is legal site or a phishing site.	High	keerthi	1
Sprint-4	Events	USN-7	This model needs the capability of retrieving and displaying accurate result for a website.	High	krithika	1

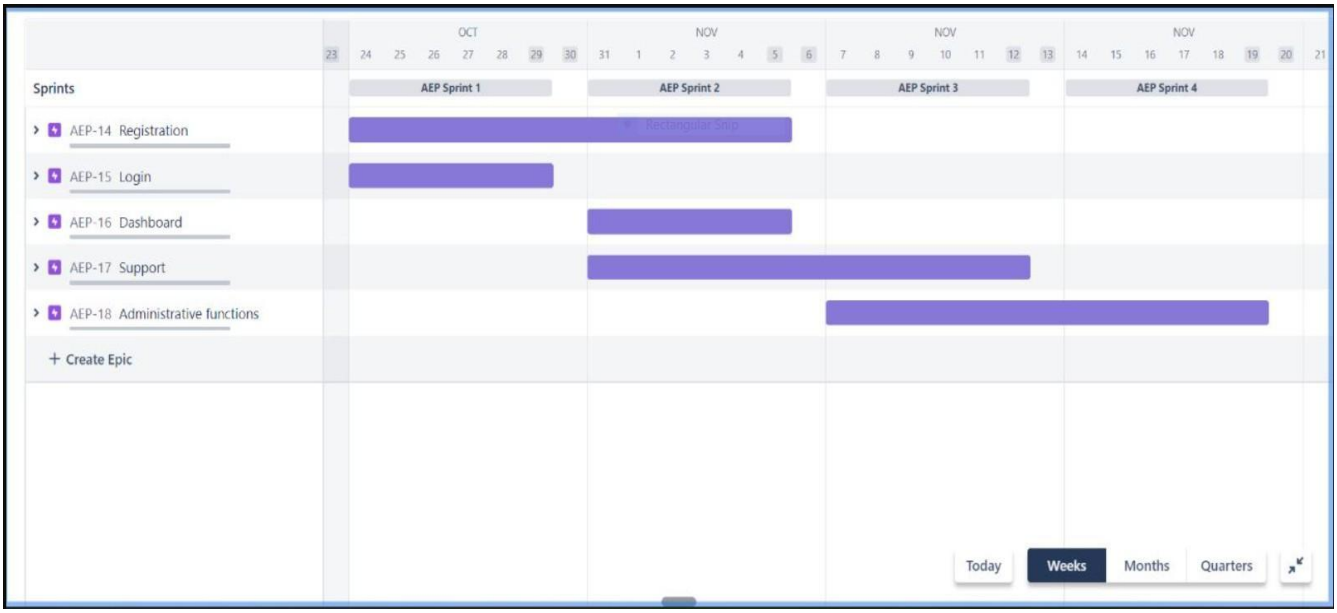
6. PROJECT PLANNING & SCHEDULING

6.1SPRINT PLANNING & ESTIMATION

6.2Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	15	6 Days	31 Oct 2022	05 Nov 2022	15	05 Nov 2022
Sprint-3	15	6 Days	07 Nov 2022	12 Nov 2022	15	12 Nov 2022
Sprint-4	15	6 Days	14 Nov 2022	19 Nov 2022	15	19 Nov 2022

6.3 REPORT FROM JIRA



7. CODING & SOLUTIONING

7.1 Feature 1

- Analyzed university admission statistics
- Developed tools for matching university (in percentile) using CGPA,GRE (Verbal, Quantitative, Analytical Writing) scores
- Languages : Python

- Tools/IDE : Anaconda(Jupyter notebook)
- Libraries : Recommendation

[1]:
```
#importing required libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from sklearn import metrics
import warnings
warnings.filterwarnings('ignore')
```

[2]:
```
#Loading data into dataframe

data = pd.read_csv("phishing.csv")
data.head()
```

t[2]:

	Index	UsingIP	LongURL	ShortURL	Symbol@	Redirecting//	PrefixSuffix-	SubDomains	HTTPS	DomainRegLen	...	UsingPopupWindow	IframeRedirection
0	0	1	1	1	1	1	-1	0	1	-1	...	1	1
1	1	1	0	1	1	1	-1	-1	-1	-1	...	1	1
2	2	1	0	1	1	1	-1	-1	-1	1	...	1	1
3	3	1	0	-1	1	1	-1	1	1	-1	...	-1	1
4	4	-1	0	-1	1	-1	-1	1	1	-1	...	1	1

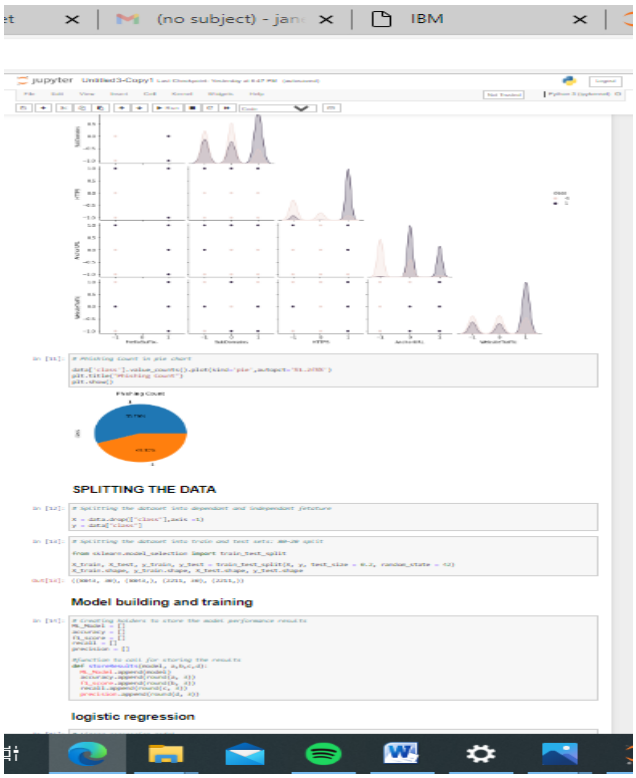
5 rows × 32 columns

[3]:
```
#Shape of dataframe

data.shape
```

t[3]:
```
(11054, 32)
```

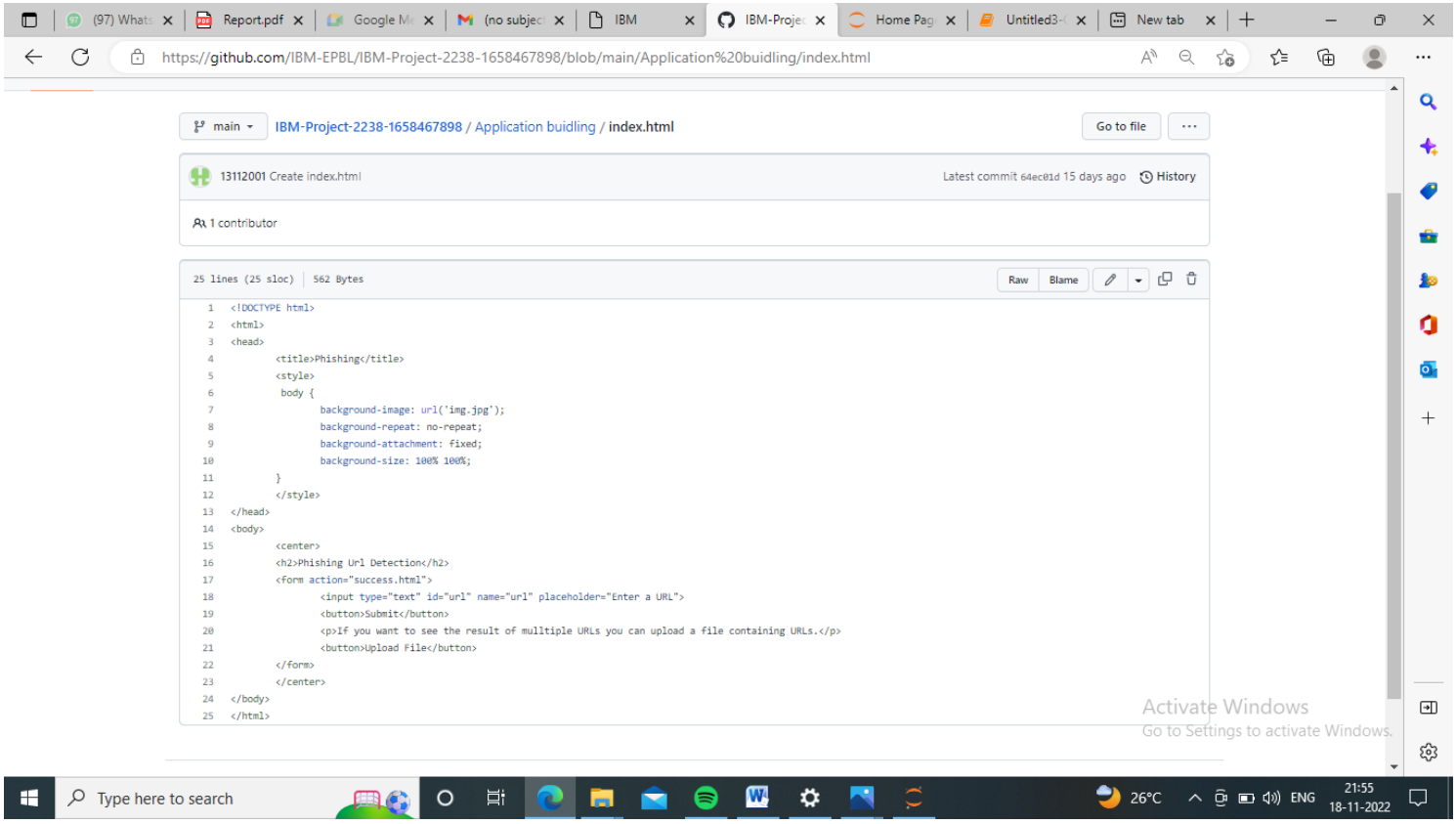
[4]:
```
#Listing the features of the dataset

data.columns
```

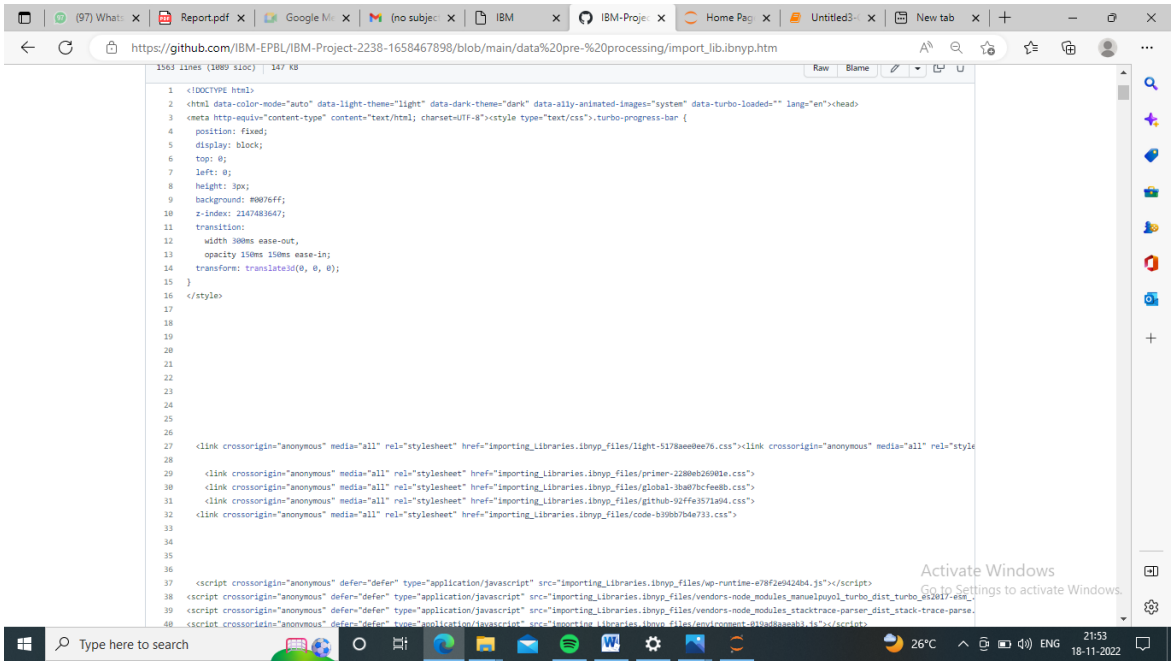
t[4]:
```
Index(['Index', 'UsingIP', 'LongURL', 'ShortURL', 'Symbol@', 'Redirecting//',
       'PrefixSuffix-', 'SubDomains', 'HTTPS', 'DomainRegLen', 'Favicon',
       'NonStdPort', 'HTTPSDomainURL', 'RequestURL', 'AnchorURL',
       'LinksInScriptTags', 'ServerFormHandler', 'InfoEmail', 'AbnormalURL',
       'WebsiteForwarding', 'StatusBarCust', 'DisableRightClick',
       'UsingPopupWindow', 'IframeRedirection', 'AgeofDomain', 'DNSRecording',
       'WebsiteTraffic', 'PageRank', 'GoogleIndex', 'LinksPointingToPage',
       'StatsReport', 'class'],
      dtype='object')
```



7.2 Feature 2

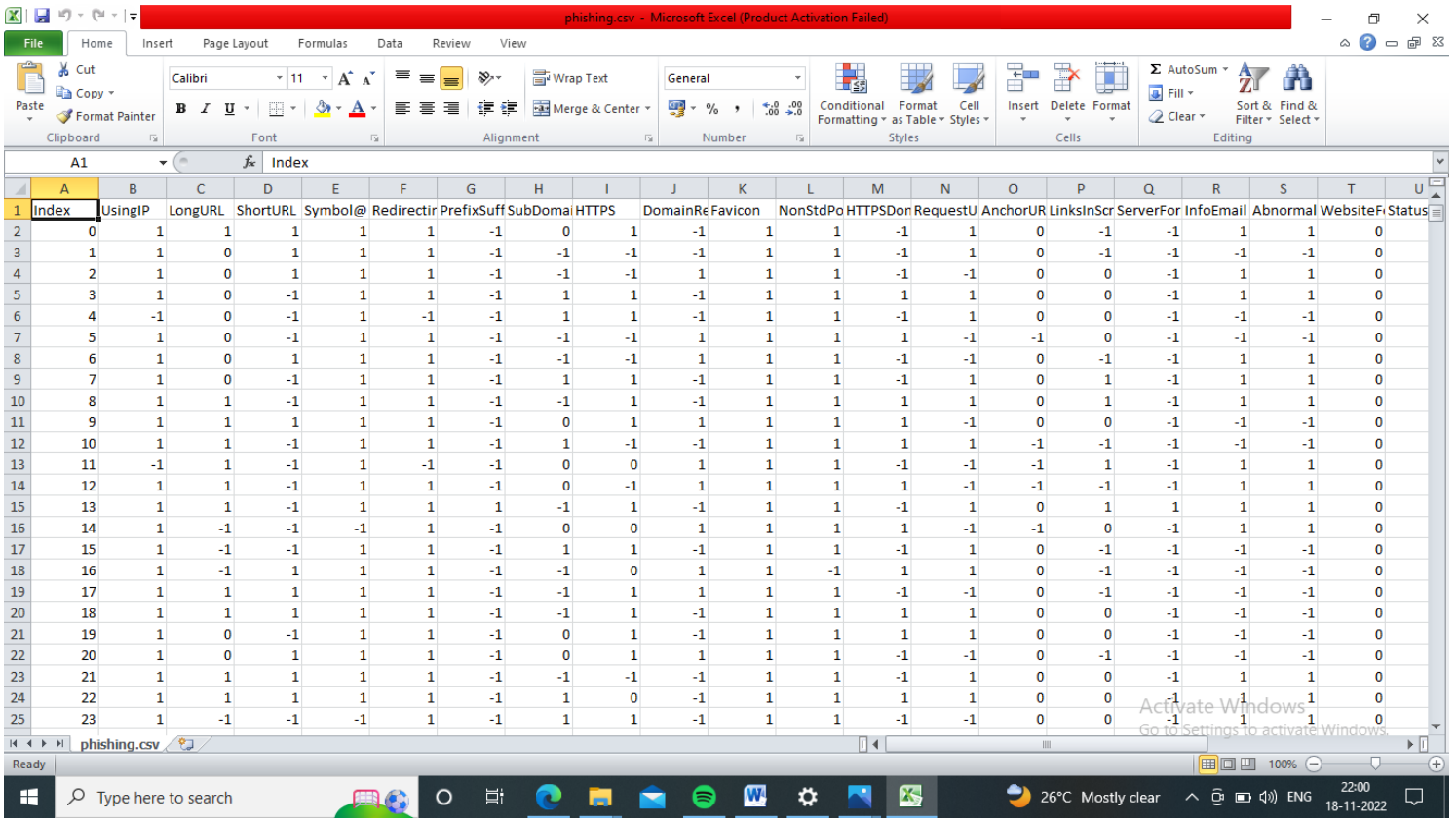
Index





Chance

7.3 Database Schema



8. TESTING

8.1 Test Cases

	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Result	Status	Comments	Automation(Y/N)	BID	Executed By
Lo	Functional	Home Page	Verify user is able to see the Landing Page when user can type the URL in the box		1.Enter URL and click go 2.Type the URL 3.Click on the URL processing or not	https://phishing-shield.herokuapp.com/	Should Display the Webpage	Working as expected	Pass		N		Janet rajajothi
Lo	UI	Home Page	Verify the UI elements is Responsive		1. Enter URL and click go 2. Type or copy paste the URL 3. Check whether the button is responsive or not 4. Reload and Test Simultaneously	https://phishing-shield.herokuapp.com/	Should Wait for Response and then gets Acknowledge	Working as expected	Pass		N		Krithika
Lo	Functional	Home page	Verify whether the link is legitimate or not		1. Enter URL and click go 2. Type or copy paste the URL 3. Check the website is legitimate or not 4. Observe the results	https://phishing-shield.herokuapp.com/	User should observe whether the website is legitimate or not.	Working as expected	Pass		N		Keerthi
Lo	Functional	Home Page	Verify user is able to access the legitimate website or not		1. Enter URL and click go 2. Type or copy paste the URL 3. Check the website is legitimate or not 4. Continue if the website is legitimate or be cautious if it is not legitimate.	https://phishing-shield.herokuapp.com/	Application should show that Safe Webpage or Unsafe.	Working as expected	Pass		N		Chandhini
Lo	Functional	Home Page	Testing the website with multiple URLs		1. Enter URL (https://phishing-shield.herokuapp.com/) and click go 2. Type or copy paste the URL to test 3. Check the website is legitimate or not 4. Continue if the website is secure or be cautious if it is not secure	1. https://sivanesan03.github.io/welcome 2. totalprod.com 3. https://www.imsec.edu 4. salescript.info 5. https://sites.google.com/	User can able to identify the websites whether it is secure or not	Working as expected	Pass		N		Keerthi

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Web Phishing Detection] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	10	2	4	20	36
Not Reproduced	0	0	1	0	1
Skipped	0	0	0	0	0
Won't Fix	0	0	2	1	3
Totals	23	9	12	25	60

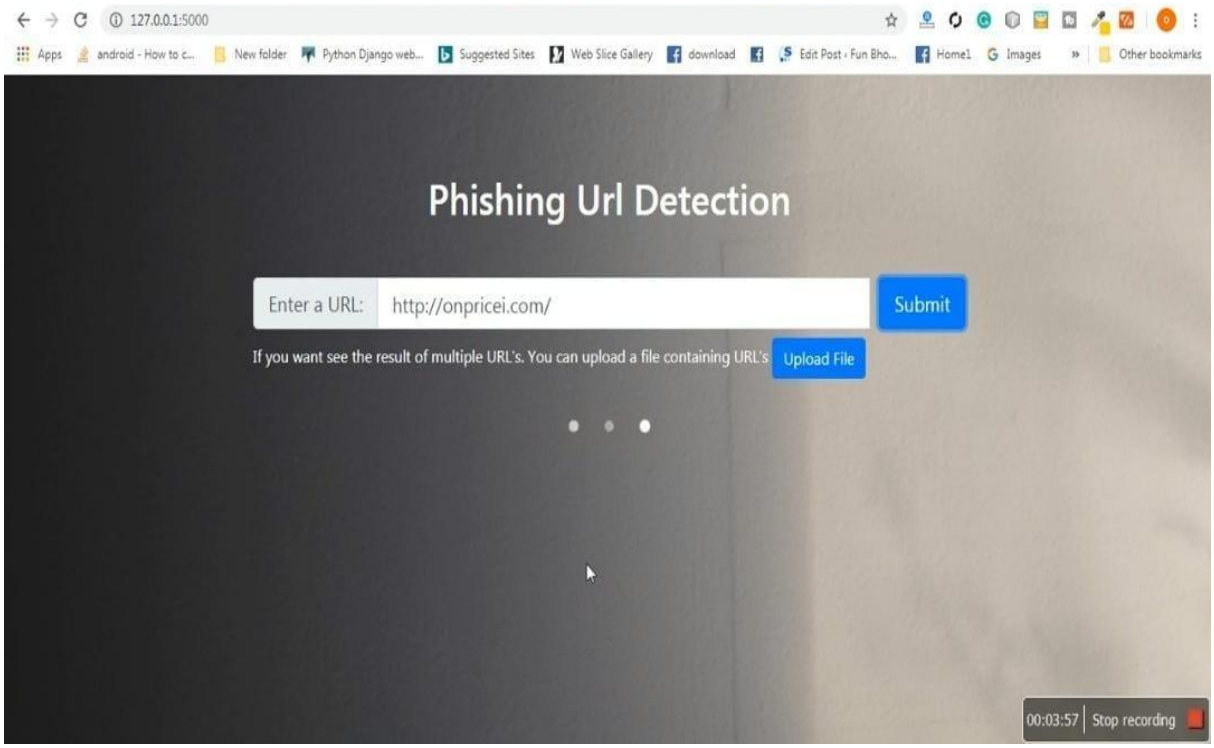
3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	10	0	0	10
Client Application	50	0	0	50
Security	5	0	0	4
Outsource Shipping	3	0	0	3
Exception Reporting	10	0	0	9
Final Report Output	10	0	0	10
Version Control	4	0	0	4

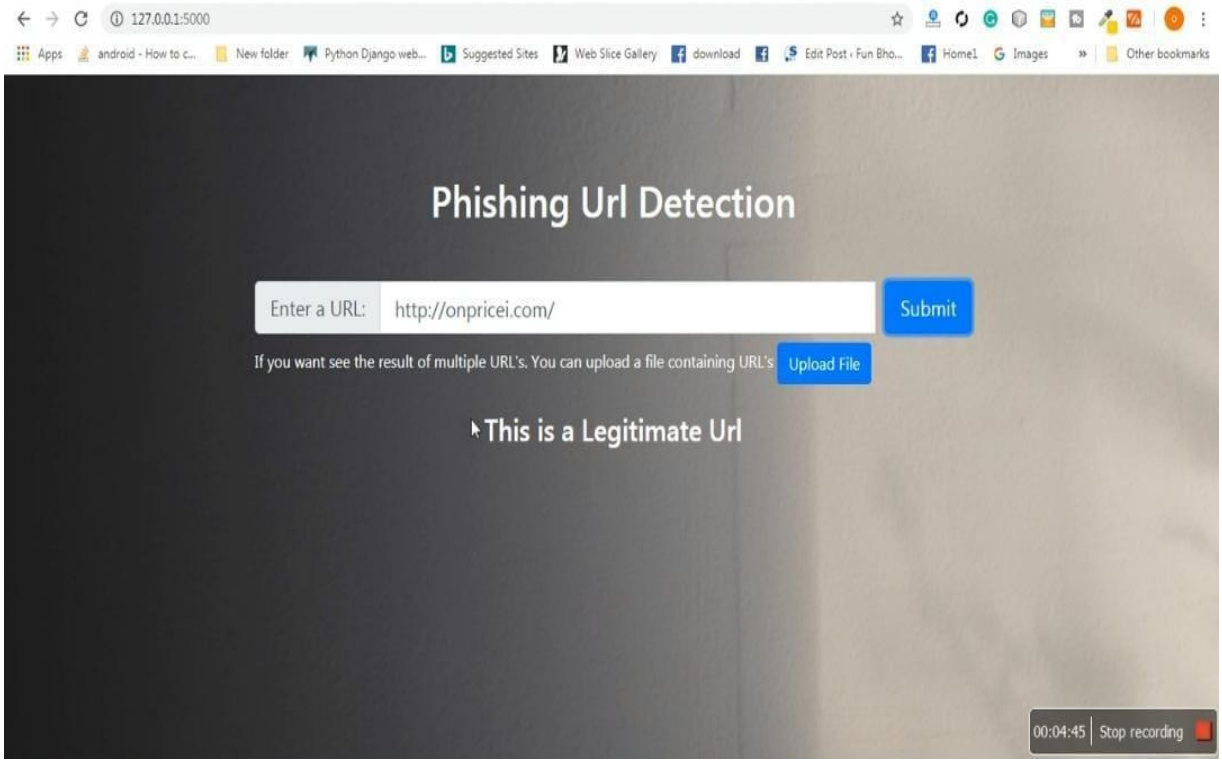
9.RESULTS

9.1 Performance Metrics



Time consuming huge number of features consuming
Consuming memory non standard classifier

Time consuming because this technique has many layers to make the final results



10 ADVANTAGES

Methods based on bag of words ,build secure connection between user's mail transfer agent(MTA) and mail user agent(MUA)
Provide clear idea about the effective algorithms provide clear idea about the effective level of each classifier on phishing email detection.
Hybrid system high level accuracy by take the advantages of many classifiers
Classifiers model based features high level of accuracy create new type of features like markov features.
Clustering of phishing email fast in classification process.

DISADVANTAGES

Time consuming huge number of features consuming
Consuming memory non standard classifier
Time consuming because this technique has many layers to make the final results
Huge number of features many algorithm for classification which mean time consuming high cost.
Need feed continuously.

11 .CONCLUSION

The proposed study emphasized the phishing technique in the context of classification, where phishing website is considered to involve automatic categorization of websites into a predetermined set of class values based on several features and the class variable. The ML based phishing techniques depend on website functionalities to gather information that can help classify websites for detecting phishing sites. The problem of phishing cannot be eradicated, nonetheless can be reduced by combating it in two ways, improving targeted anti-phishing procedures and techniques and informing the public on how fraudulent phishing websites can be detected and identified. To combat the ever evolving and complexity of phishing attacks and tactics, ML anti-phishing techniques are essential. Authors employed LSTM technique to identify malicious and legitimate websites. A crawler was developed that crawled 7900 URLs from AlexaRank portal and also employed Phishtank dataset to measure the efficiency of the proposed URL detector. The outcome of this study reveals that the proposed method presents superior results rather than the existing deep learning methods. A total of 7900 malicious URLS were detected using the proposed URL detector. It has achieved better accuracy and F1—score with limited amount of time. The future direction of this study is to develop an unsupervised deep learning method to generate insight from a URL. In addition, the study can be extended in order to generate an outcome for a larger network and protect the privacy of an individual.

12. FUTURE SCOPE

The future scope of this project is very broad. Few of them are:

- Phishing is a considerable problem differs from the other security threats such as intrusions and Malware which are based on the technical security holes of the network systems. The weakness point of any network system is its Users.

- Phishing attacks are targeting these users depending on the trikes of social engineering. Despite there are several ways to carry out these attacks, unfortunately the current phishing detection techniques cover some attack vectors like email and fake websites. Therefore, building a specific limited scope detection system will not provide complete protection from the wide phishing attack vectors.

- This paper develops detection system with a wide protection scope using URL features only which is relying on the fact that users directly deal with URLs to surf the internet and provides a good approach to detect malicious URLs as proved by

previous studies. Additionally, Anti-phishing solutions can be positioned at different levels of attack flow where most researchers are focusing on client side solutions which turn to add more processing overhead at the client side and lead to losing the trust and satisfaction of the users

- . Nowadays many organizations make centralized protection of spam filtering. This paper proposes a system which can be integrated into such process in order to increase the detection performance in a real time. The simulation results of the proposed system showed a phishing URLs detection accuracy with 93% and provided online process of a single URL in average time of 0.12 second.
  .

13. APPENDIX

13.1 Source Code

```
import pickle
from flask import Flask , request, render_template
from math import ceil
app = Flask(__name__)
model = pickle.load(open("model.pkl","rb"))

@app.route('/')
def index():
        return render_template('index.html')

@app.route('/predict',methods = ['GET','POST'])
def admin():
    gre=(eval(request.form["gre"])-290)/(340-290)
    tofl=(eval(request.form["tofl"])-92)/(120-92)
    rating=(eval(request.form["rating"])-1.0)/4.0
    sop=(eval(request.form["sop"])-1.0)/4.0
    lor=(eval(request.form["lor"])-1.0)/4.0
    cgpa=(eval(request.form["cgpa"])-6.7)/(10.0-6.7)
    research=request.form["research"]
    if (research=="Yes"):
        research=1
    else:
        research=0
    preds=[[gre,tofl,rating,sop,lor,cgpa,research]]
    xx=model.predict(preds)
    if (xx>0.5):
        return render_template("chance.html",p=str(ceil(xx[0]*100))+"%")
    return render_template("nochance.html")
if __name__ == '__main__':
    app.run(debug = False, port=4000)
```

13.2 Github & Project Demo Link

Github Link:[IBM-EPBL/IBM-Project-2238-1658467898: Web Phishing Detection (github.com)](https://github.com/IBM-EPBL/IBM-Project-2238-1658467898)

Project Demo Link:

https://drive.google.com/file/d/1n8NvAPHqv_BCv_BrfURefZrUlc06NyM3/view?usp=drivesdk